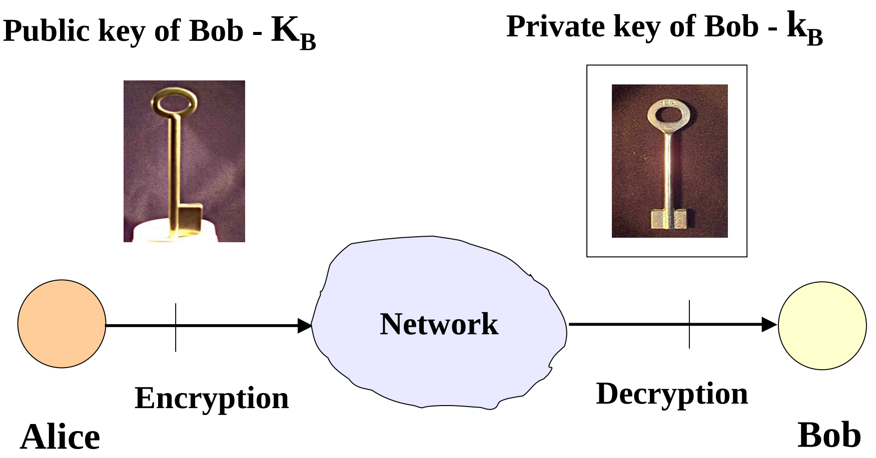
ECE297:11 Lecture 12

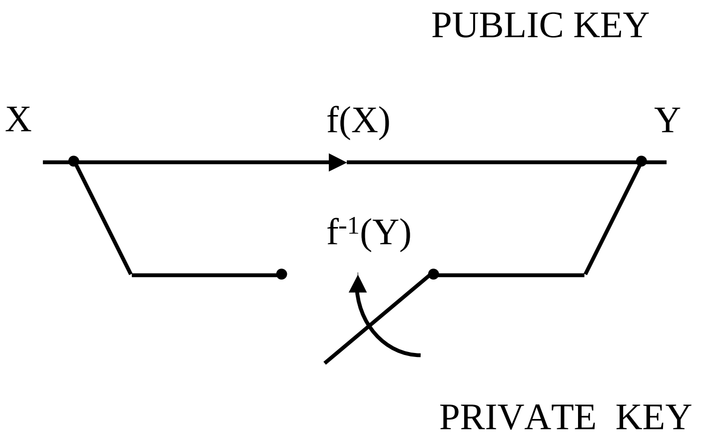
RSA – Genesis, operation & security

Public Key (Asymmetric) Cryptosystems



Trap-door one-way function

Whitfield Diffie and Martin Hellman
"*New directions in cryptography*," 1976



Professional (NSA) vs. amateur (academic) approach to designing ciphers

1. Know how to break Russian ciphers
2. Use only well-established proven methods
3. Hire 50,000 mathematicians
4. Cooperate with an industry giant
5. Keep as much as possible secret

1. Know nothing about cryptology
2. Think of revolutionary ideas
3. Go for skiing
4. Publish in "Scientific American"
5. Offer a $100 award for breaking the cipher

Challenge published in Scientific American

1977

Ciphertext:

9686 9613 7546 2206 1477 1409 2225 4355
8829 0575 9991 1245 7431 9874 6951 2093
0816 2982 2514 5708 3569 3147 6622 8839
8962 8013 3919 9055 1829 9451 5781 5145

Public key:

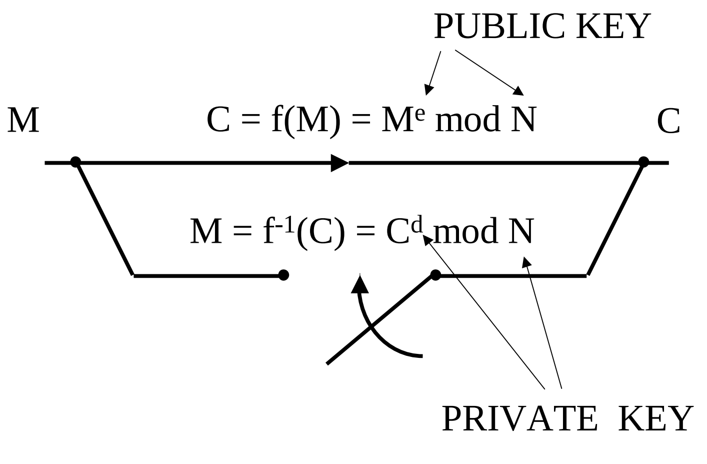
N = 114381625757 88886766923577997614
661201021829672124236256256184293
5706935245733897830597123563958705
058989075147599290026879543541

e = 9007

(**129 decimal digits**)

Award 100 $

RSA as a trap-door one-way function



$N = P \cdot Q$ P, Q - large prime numbers

$e \cdot d \equiv 1 \mod ((P-1)(Q-1))$

RSA keys

PUBLIC KEY **PRIVATE KEY**

$$\{\, e, N \,\} \quad \substack{\longleftarrow \\ \not\longrightarrow} \quad \{\, d, P, Q \,\}$$

$N = P \cdot Q$ P, Q - large prime numbers

$$e \cdot d \equiv 1 \mod ((P-1)(Q-1))$$

Why does RSA work? (1)

$$M' = C^d \bmod N = (M^e \bmod N)^d \bmod N \stackrel{?}{=} M$$

decrypted message original message

$$e \cdot d \equiv 1 \mod ((P-1)(Q-1))$$

$$\Updownarrow$$

$$e \cdot d \equiv 1 \mod \varphi(N)$$

Euler’s totient function

Euler’s totient (phi) function (1)

φ(N) - number of integers in the range from 1 to N-1 that are relatively prime with N

Special cases:

1. P is prime

$$\varphi(P) = P-1$$

Relatively prime with P: 1, 2, 3, …, P-1

2. N = P · Q   P, Q are prime

$$\varphi(N) = (P-1) \cdot (Q-1)$$

Relatively prime with N: {1, 2, 3, …, P·Q-1} – {P, 2P, 3P, …, (Q-1)P} – {Q, 2Q, 3Q, …, (P-1)Q}

Euler’s totient (phi) function (2)

Special cases:

3. N = P^2 P is prime

$$\varphi(N) = P \cdot (P-1)$$

Relatively prime with N: {1, 2, 3, …, P^2-1} – {P, 2P, 3P, …, (P-1)P}

In general

If $N = P_1^{e1} \cdot P_2^{e2} \cdot P_3^{e3} \cdot \ \ldots \ \cdot P_t^{et}$

$$\varphi(N) = \prod_{i=1}^{t} P_i^{ei-1} \cdot (P_i - 1)$$

Euler's Theorem

Leonard Euler, 1707-1783

$$\forall_{a:\ \gcd(a,\,N)=1} \quad \mathbf{a^{\varphi(N)} \equiv 1 \pmod{N}}$$

Euler's Theorem - Justification (1)

For N=10

R = {1, 3, 7, 9}

Let a=3

S = { 3·1 mod 10,
3·3 mod 10, 3·7 mod 10,
3·9 mod 10 }
= {3, 9, 1, 7}

For arbitrary N

$R = \{x_1, x_2, \ldots, x_{\varphi(N)}\}$

Let us choose arbitrary a, such that gcd(a, N) = 1

$S = \{a \cdot x_1 \bmod N, a \cdot x_2 \bmod N, \ldots, a \cdot x_{\varphi(N)} \bmod N\}$

= rearranged set R

Euler's Theorem - Justification (2)

For N=10

R = S

$x_1 \cdot x_2 \cdot x_3 \cdot x_4 \equiv$
$(a \cdot x_1) \cdot (a \cdot x_2) \cdot (a \cdot x_3) \cdot (a \cdot x_4) \bmod N$

$x_1 \cdot x_2 \cdot x_3 \cdot x_4 \equiv$
$a^4 \cdot x_1 \cdot x_2 \cdot x_3 \cdot x_4 \bmod N$

$a^4 \equiv 1 \pmod N$

For arbitrary N

R = S

$$\prod_{i=1}^{\varphi(N)} x_i \equiv \prod_{i=1}^{\varphi(N)} a \cdot x_i \pmod N$$

$$\prod_{i=1}^{\varphi(N)} x_i \equiv a^{\varphi(N)} \cdot \prod_{i=1}^{\varphi(N)} x_i \pmod N$$

$$a^{\varphi(N)} \equiv 1 \pmod N$$

Why does RSA work? (2)

$M' = C^d \bmod N = (M^e \bmod N)^d \bmod N =$

$= M^{e \cdot d} \bmod N = \left| \begin{array}{l} e \cdot d \equiv 1 \bmod \varphi(N) \\ e \cdot d = 1 + k \cdot \varphi(N) \end{array} \right| =$

$= M^{1+k \cdot \varphi(N)} \bmod N = M \cdot (M^{\varphi(N)})^k \bmod N =$

$= M \cdot (M^{\varphi(N)} \bmod N)^k \bmod N =$

$= M \cdot 1^k \bmod N = M$

Rivest estimation - 1977

The best known algorithm for factoring a
129-digit number requires:

40 000 trilion years
$= 40 \cdot 10^{15}$ years

assuming the use of a supercomputer
being able to perform

1 multiplication of 129 decimal digit numbers in 1 ns

Rivest's assumption translates to the delay of a single logic gate ≈ 10 ps

Estimated age of the universe: 100 bln years = 10^{11} years

Early records in factoring large numbers

Years	Number of decimal digits	Number of bits	Required computational power (in MIPS-years)
1974	45	149	0.001
1984	71	235	0.1
1991	100	332	7
1992	110	365	75
1993	120	398	830

How to factor for free?

A. Lenstra & M. Manasse, 1989

- **Using the spare time of computers, (otherwise unused)**
- **Program and results sent by e-mail (later using WWW)**

Practical implementations of attacks

Factorization, RSA

Year	**Number of bits of N**	**Number of decimal digits of N**	**Method**	**Estimated amount of computations**
1994	430	129	QS	5000 MIPS-years
1996	433	130	GNFS	750 MIPS-years
1998	467	140	GNFS	2000 MIPS-years
1999	**467**	**140**	**GNFS**	**8000 MIPS-years**

Breaking RSA-129

When: August 1993 - 1 April 1994, **8 months**

Who: D. Atkins, M. Graff, A. K. Lenstra, P. Leyland
+ 600 volunteers from the entire world

How: **1600 computers**
from Cray C90, through 16 MHz PC,
to fax machines

Only 0.03% computational power of the Internet

Results of cryptanalysis:

"The magic words are squeamish ossifrage"

An award of 100 $ donated to Free Software Foundation

Elements affecting the progress in factoring large numbers

- computational power
  1977-1993 increase of about 1500 times
- computer networks
  Internet
- **better algorithms**

Factoring methods

General purpose

Time of factoring depends only on the size of N

GNFS - General Number Field Sieve

QS - Quadratic Sieve

Continued Fraction Method *(historical)*

Special purpose

Time of factoring is much shorter if N or factors of N are of the special form

ECM - Elliptic Curve Method

Pollard's p-1 method

Cyclotomic polynomial method

SNFS - Special Number Field Sieve

Running time of factoring algorithms

$$L_q[\alpha, c] = \exp\left((c+o(1))\cdot(\ln q)^{\alpha}\cdot(\ln \ln q)^{1-\alpha}\right)$$

For α**=0**

$$L_q[0, c] = (\ln q)^{(c+o(1))}$$

Algorithm **polynomial** as a function of the number of bits of *q*

For α**=1**

$$L_q[1, c] = \exp((c+o(1))\cdot(\ln q))$$

Algorithm **exponential** as a function of the number of bits of *q*

For **$0 < \alpha < 1$**

Algorithm **subexponential** as a function of the number of bits of *q*

$f(n) = o(1)$ if for any positive constant $c>0$ there exist a constant $n_0>0$, such that $0 \le f(n) < c$, for all $n \ge n_0$

General purpose factoring methods

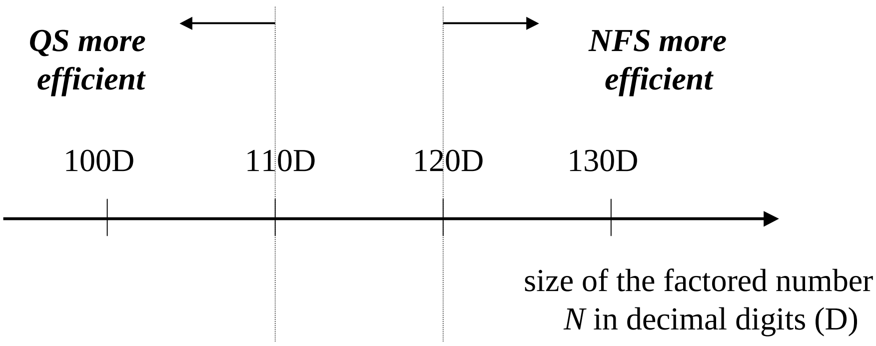
Expected running time

QS

$L_N[1/2, 1] = exp((1 + o(1))\cdot(\ln N)^{1/2}))\cdot(\ln \ln N)^{1/2})$

NFS

$L_N[1/3, 1.92] = exp((1.92 + o(1))\cdot(\ln N)^{1/3}))\cdot(\ln \ln N)^{2/3})$



RSA Challenge

RSA-100
RSA-110
RSA-120
RSA-130
RSA-140
RSA-150
RSA-160
RSA-170
RSA-180
....
RSA-450
RSA-460
RSA-470
RSA-480
RSA-490
RSA-500

Smallest unfactored number

RSA-150

Unused awards accumulate at a rate of $1750 / quarter

Factoring 512-bit number

512 bits = 155 decimal digits
old standard for key sizes in RSA

17 March - *22 August 1999*

Group of Herman te Riele
Centre for Mathematics and Computer Science
(CWI), Amsterdam

First stage 2 months

168 workstations SGI and Sun, 175-400 MHz
120 Pentium PC, 300-450 MHz, 64 MB RAM
4 stations Digital/Compaq, 500 MHz

Second stage

Cray C916 - 10 days, 2.3 GB RAM

TWINKLE
"The Weizmann INstitute Key Locating Engine"

Adi Shamir, Eurocrypt, May 1999
CHES, August 1999

Electrooptical device capable to speed-up the first phase of factorization from 100 to 1000 times

If ever built it would increase the size of the key that can be broken from 100 to 200 bits

Cost of the device (assuming that the prototype was earlier built) - $5000

Recommended key sizes for RSA

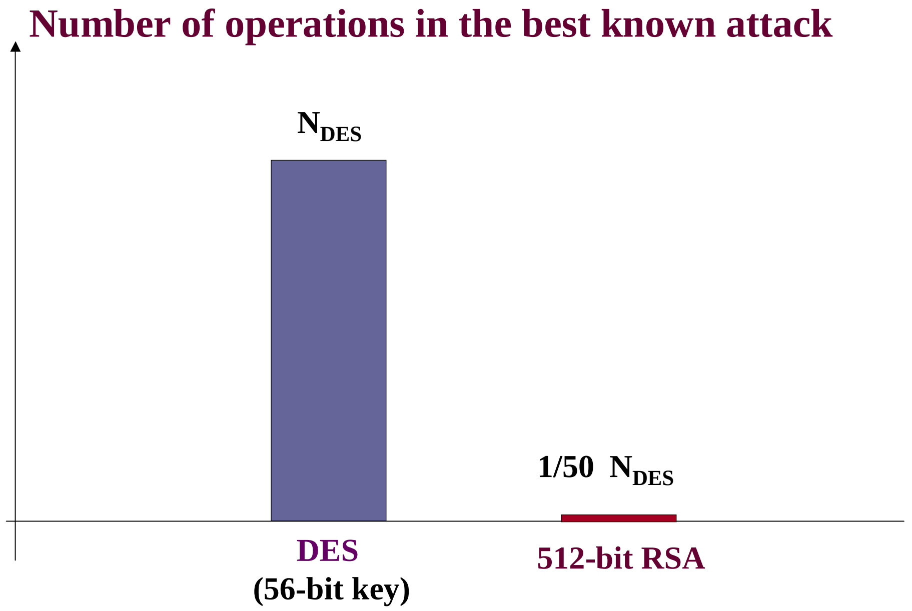
Old standard:	
Individual users	~~**512 bits**~~ ~~(155 decimal digits)~~
New standard:	
Individual users	**768 bits** (231 decimal digits)
Organizations (short term)	**1024 bits** (308 decimal digits)
Organizations (long term)	**2048 bits** (616 decimal digits)

Keylengths in public key cryptosystems that provide the same level of security as AES and other secret-key ciphers

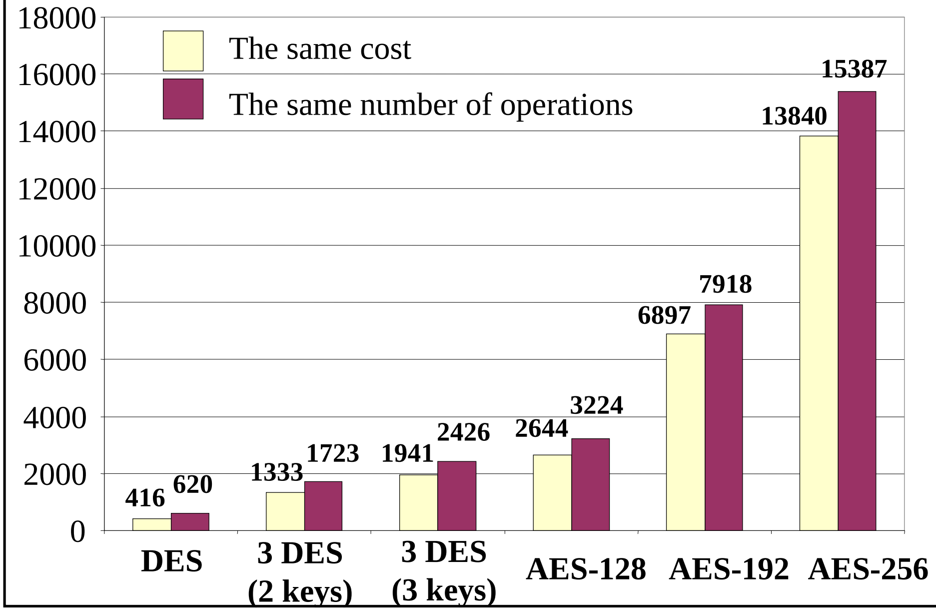
Arjen K. Lenstra, Eric R. Verheul
„***Selecting Cryptographic Key Sizes***"
Journal of Cryptology

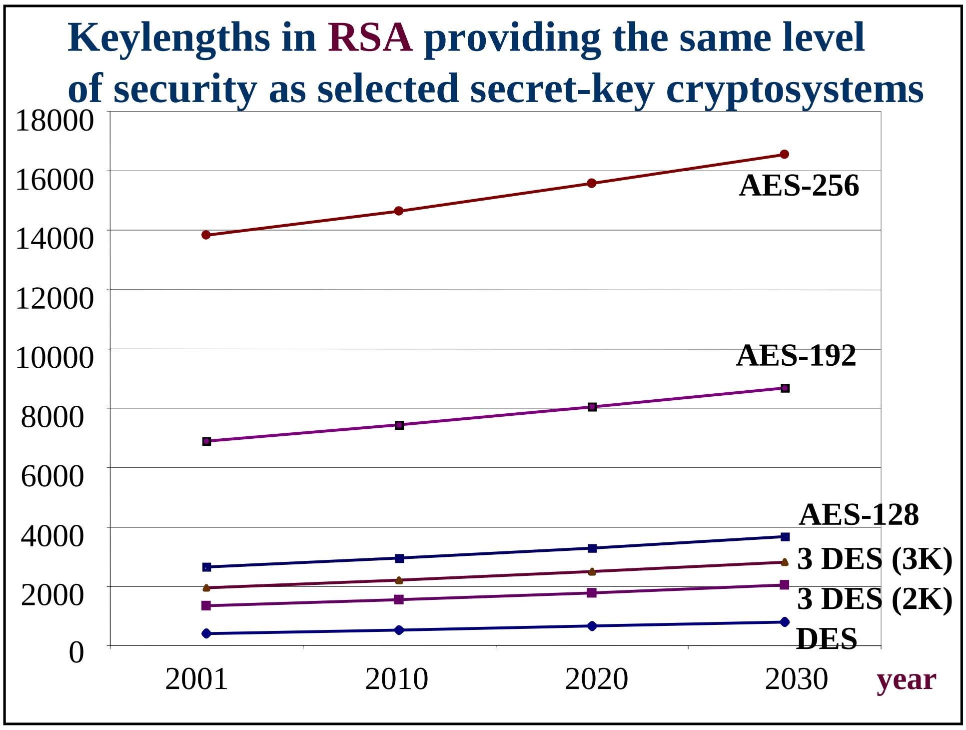
Arjen K. Lenstra
„*Unbelievable Security:* ***Matching AES Security Using Public Key Systems***"
ASIACRYPT' 2001

RSA vs. DES: Resistance to attack



Keylengths in RSA providing the same level of security as selected secret-key cryptosystems





Practical progress in factorization

March 2002, Financial Cryptography Conference

Nicko van Someren, CTO nCipher Inc.
announced that his company developed software
capable of breaking 512-bit RSA key
within **6 weeks**
using computers available in a single office

Bernstein’s Machine (1)

Fall 2001

Daniel Bernstein, professor of mathematics
at University of Illinois in Chicago
submits a grant application to NSF
and publishes fragments of this application
as an article on the web

D. Bernstein, ***Circuits for Integer Factorization: A Proposal***

http://cr.yp.to/papers.html#nfscircuit

Bernstein’s Machine (2)

March 2002

- Bernstein’s article “discovered” during *Financial Cryptography Conference*
- Informal panel devoted to analysis of consequences of the Bernstein’s discovery
- Nicko Van Someren (nCipher) estimates that machine costing $ **1 bilion** is able to break **1024-bit RSA** within **several minuts**

Bernstein's Machine (3)

March 2002

- **alarming voices** on e-mailing discussion lists calling for revocation of all currently used 1024-bit keys

- **sensational articles** in newspapers about Bernstein's discovery

Bernstein's Machine (4)

April 2002

Response of the RSA Security Inc.:

Error in the estimation presented at the conference;
according to formulas from the Bernstein's article
machine costing
$ 1 billion is able to break
1024-bit RSA within
10 billion x **several minuts = tens of years**

According to estimations of Lenstra i Verheul, machine
breaking **1024-bit RSA** within **one day**
would cost **$ 160 billion** in 2002

Bernstein's Machine (5)

Carl Pomerance, Bell Labs:

„…fresh and fascinating idea...”

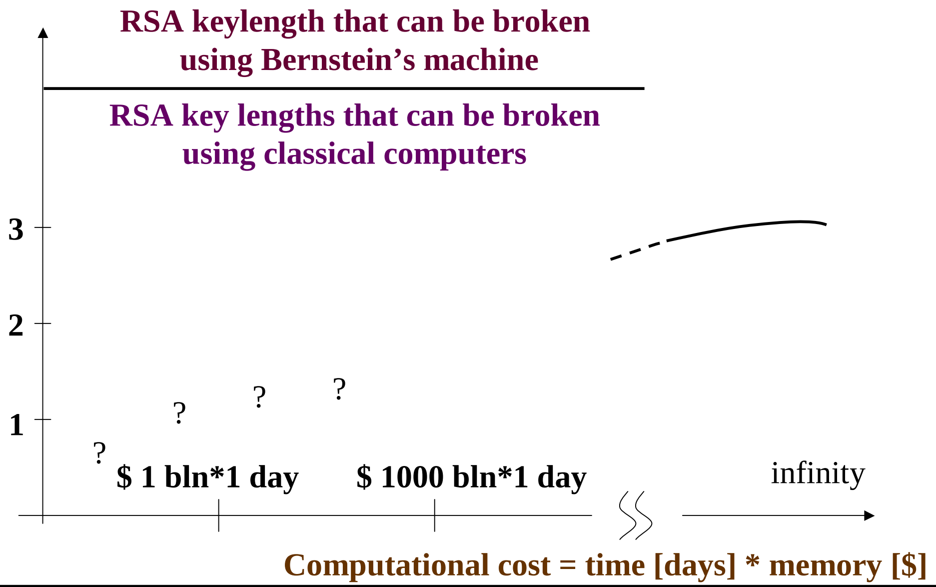
Arjen Lenstra, Citibank & U. Eindhoven:

„…I have no idea what is this all fuss about...”

Bruce Schneier, Counterpane:

„ ... enormous improvements claimed are more a result of redefining efficiency than anything else...”

Bernstein's Machine (6)



RSA Challange

Lentgh of N in bits	Length of N in decimal digits	Award for factorization
576	174	$10,000
640	193	$20,000
704	212	$30,000
768	232	$50,000
896	270	$75,000
1024	309	$100,000
1536	463	$150,000
2048	617	$200,000

Estimation of RSA Security Inc. regarding the number and memory of PCs necessary to break RSA-1024

Attack time: 1 year

Single machine: PC, 500 MHz, 170 GB RAM

Number of machines: 342,000,000